IN THE CLAIMS:

Please amend the claims as follows. This listing of claims will replace all prior versions, and listings, of claims in the present application:

1. (Currently amended) A method of transmitting executable software from a server to a client computer, the method comprising:

segmenting each of a plurality of software applications into a collection of executable blocks, to form a plurality of collections of executable blocks;

forming ~~an InitBlock Bundle~~a startup block bundle comprising blocks executable during initialization of the plurality of software applications, at least one executable block from each of the plurality of software applications being included in the ~~InitBlock Bundle~~startup block bundle;

sending the ~~InitBlock Bundle~~startup block bundle to a client computer to enable the client computer to execute the plurality of software applications in a streaming mode; and

sending other blocks from the plurality of collections of executable blocks to the client computer subsequent to a start of execution of the ~~InitBlock Bundle~~startup block bundle.

2. (Currently amended) The method of claim 1 wherein:

the plurality of software applications comprise at least one software application subscribed to by a user and at least one software application not subscribed to by the user; and

the method further comprises:

monitoring execution of software applications subscribed to by the user to determine ~~an~~ a software application usage pattern; and

based on the usage pattern, sending data to the client ~~terminal~~ computer to display information about a first one of the ~~unsubscribed~~ software applications not subscribed to by the user.

3. (Currently amended) The method of claim 2 wherein:

the data to display information about a first one of the software applications not subscribed to by the user ~~unsubscribed applications~~ comprises an offer to subscribe to the first one of the software applications not subscribed to by the user~~unsubscribed applications~~.

4. (Currently amended) The method of claim 3 further comprising:

in response to the offer to subscribe, receiving data at the server indicating acceptance of the offer; and

sending data to the client ~~terminal~~ computer to enable execution of the first one of the ~~unsubscribed~~ software applications not subscribed to by the user.

5. (Currently amended) The method of claim 4 wherein:

the ~~InitBlock Bundle~~startup block bundle comprises access control data; and

sending data to enable execution comprises sending changed access control data from the server to the client computer.

6. (Original) The method of claim 5 wherein sending the changed access control data comprises automatically sending in response to a subscription request received at the server from the client computer.

7. (Currently amended) The method of claim 5 wherein the access control data comprises an encryption key enabling access to blocks of subscribed-to software applications.

8. (Currently amended) The method of claim 1 further comprising:

from each of a plurality of service providers, sending to a client terminal ~~an InitBlock Bundle~~ a startup block bundle comprising a plurality of initialization blocks;

monitoring execution of blocks in each of said ~~InitBlock Bundles~~ startup block bundles to determine a usage pattern; and

forming a new ~~InitBlock Bundle~~startup block bundle based on the usage pattern.

9. (Currently amended) The method of claim 8 wherein the new ~~InitBlock Bundle~~startup block bundle comprises executable blocks associated with software applications from different ones of the service providers.

10. (Currently amended) The method of claim 1 wherein at least one of the blocks in the ~~InitBlock Bundle~~startup block bundle is a shared block executable during the initialization phase of different ones of the software applications.

11. (Currently amended) The method of claim 1 wherein the ~~InitBlock Bundle~~startup block bundle comprises, for each of the plurality of software applications, a set of blocks

sufficient to enable execution of ~~each of the plurality of applications~~ said software application to a point when ~~the~~ said software application awaits user input.

12. (Currently amended) The method of claim 1 wherein forming the ~~InitBlock Bundle~~startup block bundle comprises: monitoring usage of a plurality of different software applications; and wherein forming the initialization block comprises forming based on the monitored usage.

13. (Currently amended) The method of claim 1 further comprising:

sending from the server to the client a plurality of key values, the key values identifying ones of the collection of executable blocks;

receiving a response at the server from the client indicating blocks identified by the key values that are already stored at the client; and wherein

sending the ~~InitBlock Bundle~~startup block bundle comprises omitting blocks stored already stored at the client.

14-18. (Canceled)

19. (Currently amended) A computer system comprising:

a database storing a plurality of ~~executable~~ software applications segmented into a plurality of executable code blocks, each said software application's plurality of code blocks comprising a set of executable initialization code blocks;

a processor operatively coupled to a network interface, to the database and to a computer readable data storage media comprising instructions to configure the processor to:

form an initialization block comprising initialization code blocks for at least two of the plurality of software applications; and

send the initialization block to a client computer operatively coupled to the network interface computer to enable the client computer to execute the plurality of software applications in a streaming mode.

20. (Currently amended) The system of claim 19 wherein the data storage media further comprises instructions to configure the processor to:

monitoring execution of initialization code blocks at the client computer to determine a usage pattern; and

forming a new ~~InitBlock Bundle~~startup block bundle based on the usage pattern.

21. (Currently amended) The system of claim 19 wherein:

the system further comprises a database comprising a plurality of user profiles, each user profile comprising security data to control usage of ones of the plurality of software applications by a respective user;

the data storage media further comprises:

______instructions to query the database of user profiles to access security data associated with a first user;

instructions to process the security data to determine application restriction data associated with the first user; and

instructions to send the application restriction data to the first client computer.

22. (Currently amended) The system of claim 21 wherein the application restriction data comprises further comprises data preventing user access to ~~the unsubscribed second~~a software application not subscribed to by the first user.

23. (Currently amended) A computer readable ~~data~~ storage ~~apparatus~~ medium storing instructions for configuring a computer to:

send to a client terminal a key value identifying a streamable block;

receive a response from the client terminal indicating whether the client terminal has a locally stored copy of the block; and

send the block to the client terminal if the client terminal does not have a locally stored copy.

24. (Currently amended) The storage medium ~~apparatus~~ of claim 23 wherein:

the instructions to send a key value further ~~comprises~~ comprise instructions to simultaneously send a group of other key values identifying other streamable blocks;

the instructions to receive a response further ~~comprises~~ comprise instructions to receive a response indicating whether the client terminal has locally stored copies of ones of the other blocks; and

the instructions to send the block further ~~comprises~~ comprise instructions to send ones of the other blocks that are not locally stored at the client.

25-48. (Canceled)

49. (New) An apparatus comprising:

means for segmenting each of a plurality of software applications into a collection of executable blocks, to form a plurality of collections of executable blocks;

means for forming a startup block bundle comprising blocks executable during initialization of the plurality of software applications, at least one executable block from each of the plurality of software applications being included in the startup block bundle;

means for sending the startup block bundle to a client computer to enable the client computer to execute the plurality of software applications in a streaming mode; and

means for sending other blocks from the plurality of collections of executable blocks to the client computer subsequent to a start of execution of the startup block bundle.